

# Java na Web



## Aplicações Web

**Montando o ambiente com HSQLDB**

**JSP com um conteúdo de banco de dados: JDBC**

**Integre JSPs com *servlets* aplicando o Modelo-Visualização-Controle**

# Java na Web

## Aplicações Web



### Montando o ambiente com HSQLDB

Faça a transferência da página da aula do arquivo “hsqldb\_1\_8\_0\_RC10.zip”, para o diretório local do computador.

Vá até este local depois de completar a transferência e faça a descompressão do arquivo.

Abra um editor de texto e cole este texto:

```
java -classpath ./lib/hsqldb.jar org.hsqldb.Server -database.0 dbmvc -dbname.0 mvc
```

Salve o arquivo com o nome “run\_server\_hsqldb.bat”

# Java na Web

## Aplicações Web



### Montando o ambiente com HSQLDB

**Abra o Windows Explorer e clique duas vezes no arquivo “run\_server\_hsqlfb.bat”.**

**Um terminal DOS será aberto e não poderá ser fechado.  
O HSQLDB no modo Web está no ar.**

**Abra um editor de texto e cole este texto:**

```
java -classpath ./lib/hsqldb.jar org.hsqldb.util.DatabaseManagerSwing
```

**Salve o arquivo com o nome “Manager.bat”**

**Abra o Windows Explorer e clique duas vezes no arquivo “Manager.bat”.**

# Java na Web

## Aplicações Web



### Montando o ambiente com HSQLDB

Um terminal DOS será aberto com um aplicativo Swing que o Gerenciador do HSQLDB.

Em “Setting Name” coloque DB;

Em “Type” altere para “HSQL Database Engine WebServer”;

Não altere o valor que está em “Driver”;

Em “URL” coloque “jdbc:hsqldb:http://localhost/mvc”;

Não altere nem o parâmetro “User” e “Password”;

Clique em OK.

Abra um editor de texto e cole este texto:

```
java -classpath ./lib/hsqldb.jar org.hsqldb.util.DatabaseManagerSwing
```

Salve o arquivo com o nome “Manager.bat”

Abra o Windows Explorer e clique duas vezes no arquivo

# Java na Web

## Aplicações Web



### Montando o ambiente com HSQLDB

```
java -classpath ./lib/hsqldb.jar org.hsqldb.Server -database.0 dbmvc -dbname.0 mvc
```

# Java na Web

## Aplicações Web



### Montando o ambiente com HSQLDB

```
CREATE TABLE AGENDA (  
  ID INTEGER NOT NULL PRIMARY KEY,  
  NOME VARCHAR(40),  
  EMAIL VARCHAR(30)  
);  
INSERT INTO AGENDA VALUES (1, 'Carlos', 'carlosfgo@gmail.com');  
INSERT INTO AGENDA VALUES (2, 'Marli', 'marliespregag@gmail.com');  
INSERT INTO AGENDA VALUES (3, 'Rodrigo', 'constantino@gmail.com');  
INSERT INTO AGENDA VALUES (4, 'Flavio', 'flaviocurte@yahoo.com.br');
```

# Java na Web

## Aplicações Web



```
<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8"%>
<%@page import="java.util.*,java.sql.*;"
%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>HSQLDB</title>
  </head>
  <body>

    <h1>HSQLDB</h1>

  <%
String driver = "org.hsqldb.jdbcDriver";
String url = "jdbc:hsqldb:http://localhost/mvc";
String usuario = "sa";
String senha = "";
```

# Java na Web

## Aplicações Web



```
Class.forName(driver);
Connection con = DriverManager.getConnection(url, usuario, senha);
Statement st = con.createStatement();
String sql = "SELECT nome, email FROM AGENDA";
ResultSet res = st.executeQuery(sql);
%>
<table border="0" bgcolor="white" width="100%">
<%
while (res.next()) {
%>
<tr> <td>
    <%=res.getString(1)%>
</td> <td>
    <%=res.getString(2)%>
</td> <tr>
<%
}
res.close(); st.close(); con.close();

%>
</table>

</body>
</html>
```

# Java na Web

## Padrão MVC



## Aplicações Web MVC

No desenvolvimento visual(antes do Java) se popularizou um estilo pouco organizado de desenvolvimento, com o código espalhado pelas rotinas de tratamento de eventos. Já para criação de aplicações web foi estabelecido um estilo mais organizado, baseado na separação das responsabilidades em: exibição de informações (Visualização), acesso a dados e execução de lógica de negócios (Modelo), e mediação entre a Visualização e o Modelo(Controlador).

# Java na Web

## Padrão MVC



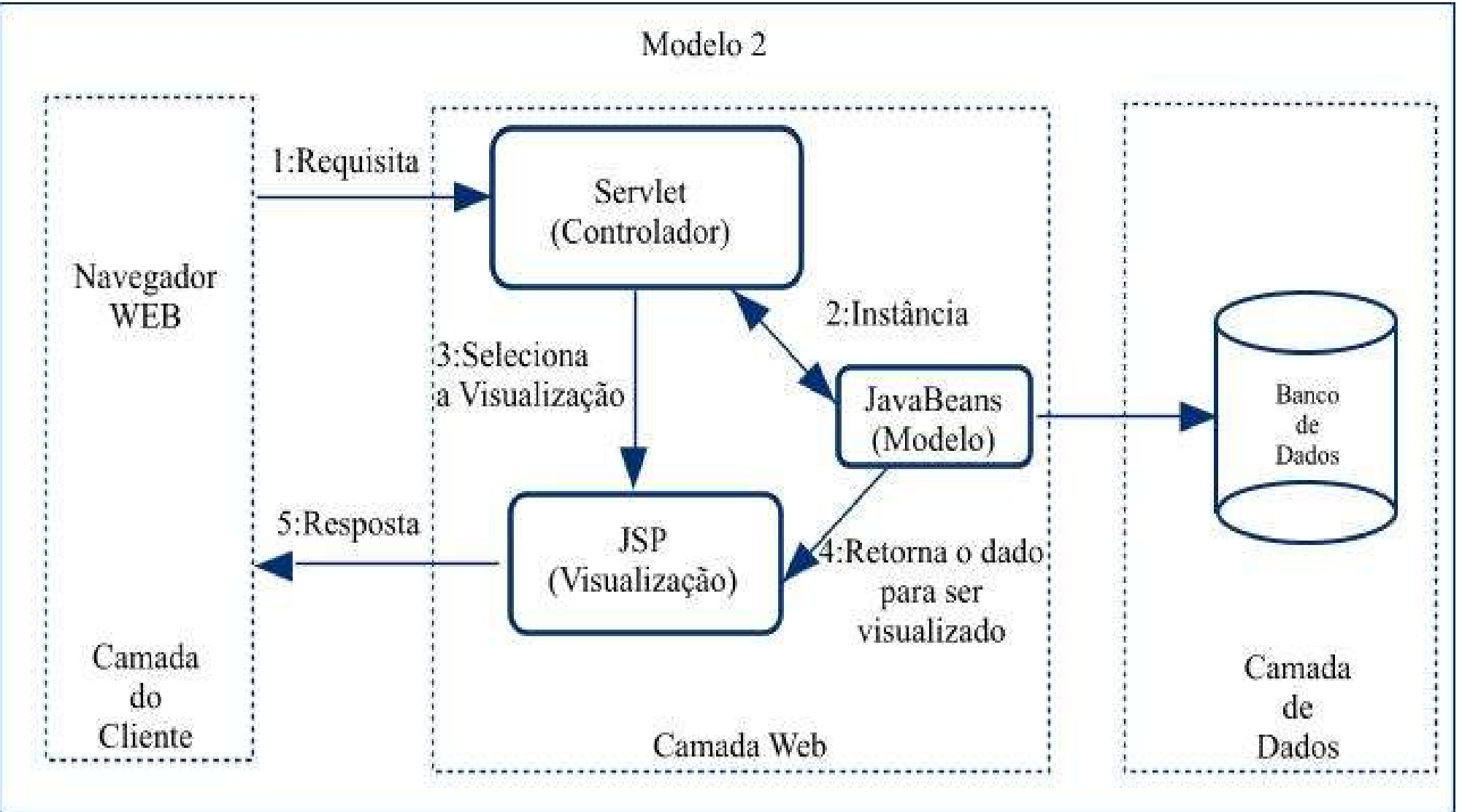
## Aplicações Web MVC

A sigla MVC vem do inglês *Model-View-Controller*(Modelo-Visualização-Controle), é um padrão MVC que oferece uma boa separação entre as responsabilidades do sistema nas tarefas de dados (modelo), o gerenciamento das tarefas (controle) e as relações entre os dados e a forma(visualização) no qual serão exibidas.

MVC é o padrão para desenvolvimento Java para Web da SUN Microsystems, criado originalmente na linguagem Smalltalk para desenvolvimento de aplicações gráficas(GUI).

# Java na Web

## Padrão MVC



# Java na Web

## Padrão MVC



## Aplicações Web MVC

O ganho para o desenvolvimento é separar melhor as responsabilidades, diminuir código JSP e reusabilidade. Neste padrão MVC, é utilizada a arquitetura de implementação com servlet, JavaBeans e JSP. O Controle fica a cargo de um servlet que gerencia o fluxo da aplicação Web, o Modelo é delegado para um JavaBeans e a Visualização é de responsabilidade de uma JSP, que produz uma saída em HTML para o navegador Web do cliente.